

REGULATING UNUSED/INACTIVE RESOURCES IN PROGRAMMABLE LOGIC
DEVICES FOR STATIC POWER REDUCTION

Kevin T. Look

Michael J. Hart

Tim Tuan

Kameswara K. Rao

Robert O. Conn

Related Application

[0001] The present application is a continuation-in-part of U.S. Patent Application Serial No. 10/666,669 filed by Tim Tuan, Kameswara K. Rao and Robert O. Conn on September 19, 2003, which is incorporated herein in its entirety.

FIELD OF THE INVENTION

[0002] The present invention relates to the regulation of the supply voltage provided to unused and/or inactive blocks in a programmable logic device to achieve lower static power consumption. More specifically, the present invention relates to selectively reducing the operating voltage of various sections of an integrated circuit device in order to reduce the leakage current and/or increase the performance of the device.

RELATED ART

[0003] Improved performance in integrated circuits requires technology scaling. As technology is scaled, the physical dimensions of transistors, such as gate oxide thickness and transistor gate length, are reduced. The operating voltage of the transistor must scale to maintain an acceptable electric field across the gate oxide to maintain acceptable reliability. Lowering the operating voltage requires that the threshold voltage of the transistor be reduced. Transistor off-state leakage current consists of sub-threshold leakage, gate direct tunnel current, and band-to-band tunnel leakage current. The relationship of transistor off-state leakage currents with respect to device scaling is shown by Inukai, et al., in "Boosted Gate MOS (BGMOS): Device/Circuit Cooperation Scheme to

Achieve Leakage-Free Giga-Scale Integration", IEEE Custom Integrated Circuit Conference, 19-2-1, 2000.

[0004] Inukai et al. teach that the sub-threshold leakage of the transistor becomes increasing large from one technology node to the next technology. At the same time, the gate leakage current of the transistor also increases as a result of reduction in gate oxide thickness. It has been reported that the gate leakage current increases by an order of magnitude for each 2 Angstrom reduction in gate oxide thickness. (Hamzaoglu et al., "Circuit-Level Techniques to Control Gate Leakage for sub-100nm CMOS", ISLPED. p.60-63, Aug, 2002.) Inukai et al. project that the gate leakage current will exceed the sub-threshold leakage current and become the dominating factor in leakage at some technology node. Therefore, standby power becomes more and more problematic in technology scaling.

[0005] Circuit techniques to suppress leakage currents have been proposed in the literature. Kuroda et al. propose a VTMOS technique to suppress standby leakage current in "0.9V. 150-MHz, 10-mW, 4 mm², 2-D discrete cosine transform core processor with variable threshold-voltage (VT) scheme", IEEE Journal of Solid-State Circuits, vol. 31. pp. 1770-1779, 1996. In this VTMOS technique, the transistor is back-biased to raise the threshold voltage of the transistor in a standby mode. This technique can reduce the transistor sub-threshold leakage current, but does not reduce the gate tunneling leakage current.

[0006] Mutoh et al. propose an MTMOS technique in "I-V Power Supply High-Speed Digital Circuit Technology With Multi-Threshold Voltage CMOS", IEEE Journal of Solid-State Circuit, vol. 30, pp. 847-854, 1995. This MTMOS technique uses a transistor having a high threshold voltage (V_t) to supply a virtual V_{DD} supply voltage and a ground supply voltage to core circuitry. The high V_t transistor has the same gate oxide thickness as transistors in the core circuitry, but is less leaky. However, the MTMOS technique faces a similar problem as the VTMOS technique. That is, the MTMOS technique does not reduce gate leakage current.

[0007] Inukai et al. propose a BGMOS technique. The BGMOS technique uses a thicker oxide transistor as a power switch to shut-off the leakage path in a standby mode. However, the disadvantage of this technique is that the stored data are lost in standby mode. Inukai et al. propose adding a memory cell to store the data in standby mode. However, this proposed solution results in a significant increase in area penalty.

[0008] It would therefore be desirable to have a technique that can reduce both the gate tunneling leakage and sub-threshold leakage while maintaining the circuit state without significantly increasing the area penalty for the upcoming technology node.

[0009] Note that programmable logic devices (PLDs), such as field programmable gate arrays (FPGAs), have a significantly higher static power consumption than dedicated logic devices, such as standard-cell application specific integrated circuits (ASICs). A reason for this high static power consumption is that for any given design, a PLD only uses a subset of the available resources. The unused resources are necessary for providing greater mapping flexibility to the PLD. However, these unused resources still consume static power in the form of leakage current. Consequently, PLDs are generally not used in applications where low static power is required.

[0010] It would therefore be desirable to have a PLD having a reduced static power consumption.

SUMMARY

[0011] In accordance with one embodiment of the present invention, unused and/or inactive resources in a PLD are disabled to achieve lower static power consumption.

[0012] One embodiment of the present invention provides a method of operating a PLD, which includes the steps of enabling the resources of the PLD that are used in a particular circuit design, and disabling the resources of the PLD that are unused or inactive. The step of disabling can include de-coupling the unused or inactive resources from one or more power supply

terminals. Alternately, the step of disabling can include regulating (e.g., reducing) a supply voltage applied to the unused or inactive resources.

[0013] In accordance with one embodiment, the step of disabling can be performed in response to configuration data bits stored by the PLD. These configuration data bits can be determined during the design of the circuit to be implemented by the PLD. That is, during the design, the design software is able to identify unused resources of the PLD, and select the configuration data bits to disable these unused resources.

[0014] The step of disabling can also be performed in response to user-controlled signals. These user-controlled signals can be generated in response to observable operating conditions of the PLD. For example, if certain resources of the operating PLD are inactive for a predetermined time period, then the user-controlled signals may be activated, thereby causing the inactive resources to be disabled.

[0015] In accordance with another embodiment, a PLD includes a first voltage supply terminal that receives a first supply voltage, a plurality of programmable logic blocks, and a plurality of switch elements, wherein each switch element is coupled between one of the programmable logic blocks and the first voltage supply terminal. A control circuit coupled to the switch elements provides a plurality of control signals that selectively enable or disable the switch elements. The control circuit can be controlled by a plurality of configuration data values stored by the PLD and/or a plurality of user-controlled signals.

[0016] In an alternate embodiment, each of the switch elements can be replaced by a switching regulator. In accordance with one embodiment, the switching regulator can be a high-voltage n-channel transistor having a drain coupled to the VDD voltage supply and a source coupled to the programmable logic block. The gate of the high voltage transistor is coupled to receive a control voltage from a corresponding control circuit. The control circuit determines whether the corresponding programmable logic block is in an active or

inactive state in response to user controlled signals and/or configuration data bits. When the programmable logic block is active, the control circuit applies a high control voltage V_{BOOST} , which is greater than the V_{DD} supply voltage, to the gate of the high voltage transistor, such that the full V_{DD} supply voltage is applied to the programmable logic block. When the programmable logic block is inactive, the control circuit applies a low control voltage V_{STANDBY} , which is less than the V_{DD} supply voltage, to the gate of the high voltage transistor, such that a voltage of about one half the V_{DD} supply voltage is applied to the programmable logic block. A feedback mechanism can be employed to ensure that the voltage applied to the programmable logic block is precisely equal to one half the V_{DD} supply voltage.

[0017] The present invention will be more fully understood in view of the following description and drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] Fig. 1 is a flow diagram illustrating a conventional design flow used for PLDs.

[0019] Fig. 2 is a flow diagram illustrating a design flow for a PLD in accordance with one embodiment of the present invention.

[0020] Fig. 3 is a block diagram of a conventional PLD having four blocks, which are all powered by the same off-chip V_{DD} voltage supply.

[0021] Fig. 4 is a block diagram of a PLD that implements power-gating switch elements in accordance with one embodiment of the present invention.

[0022] Fig. 5 is a block diagram of a PLD that implements switching regulators in accordance with one embodiment of the present invention.

[0023] Fig. 6 is a block diagram of the PLD of Fig. 5, which shows switching regulators in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION

[0024] In accordance with one embodiment of the present invention, unused and inactive resources in a programmable logic device (PLD), such as a field programmable gate array (FPGA), are disabled to achieve lower static power consumption. The present invention includes both an enabling software flow and an enabling hardware architecture, which are described in more detail below. Unused resources of the PLD can be disabled when designing a particular circuit to be implemented by the PLD (hereinafter referred to as "design time"). In addition, resources of the PLD that are temporarily inactive can be disabled during operation of the PLD (hereinafter referred to as "run time").

[0025] Fig. 1 is a flow diagram 100 illustrating a conventional design flow used for PLDs. Initially, a user designs a circuit to be implemented by the PLD (Step 101). This user design is described in a high-level specification, such as Verilog or VHDL. The high-level specification is first synthesized to basic logic cells available on the PLD (Step 102). A place and route process then assigns every logic cell and wire in the design to some physical resource in the PLD (Step 103). The design is then converted into a configuration bit stream, in a manner known to those of ordinary skill in the art (Step 104). The configuration bit stream is then used to configure the device by setting various on-chip configuration memory cells (Step 105). While modern design flows may be much more complex, they all involve the basic steps defined by flow diagram 100.

[0026] In accordance with the present invention, unused resources of the PLD are identified during the design time, following the place and route process (Step 103). These unused resources are then selectively disabled during the design time. As described below, there are several ways to disable the unused resources. By selectively disabling the unused resources at design time, significant static power reduction may be achieved with no performance penalty.

[0027] Fig. 2 is a flow diagram 200 illustrating a design flow in accordance with one embodiment of the present invention. Similar steps in flow diagrams 100 and 200 are labeled with similar reference numbers. Thus, flow diagram 200 includes Steps 101-105 of flow diagram 100, which are described above. In addition, flow diagram 200 includes the step of disabling unused resources in the PLD (Step 201). This step of disabling unused resources is performed after the place and route process has been completed in Step 103, and before the configuration bit stream is generated in Step 104. As described in more detail below, the unused resources are disabled by disabling predetermined programmable logic blocks of the PLD.

[0028] In another embodiment, further power savings are obtained by disabling temporarily inactive resources of the configured PLD during run time. Often, the entire design or parts of the design are temporarily inactive for some period of time. If the inactive period is sufficiently long, it is worthwhile to disable the inactive resources to reduce static power consumption. In a preferred embodiment, the decision of when to disable a temporarily inactive resource is made by the designer. In this embodiment, the user logic is provided access to a disabling mechanism, which enables the inactive resources to be disabled dynamically.

[0029] There are a number of techniques to disable resources in a PLD. In accordance with one embodiment, the PLD is logically subdivided into a plurality of separate programmable logic blocks. As described below, each programmable logic block may comprise one or more of the resources available on the programmable logic device. Switch elements are used to couple each of the programmable logic blocks to one or more associated voltage supply terminals (e.g., V_{DD} or ground). The switch elements are controlled to perform a power-gating function, wherein unused and/or inactive programmable logic blocks are disabled (e.g., prevented from receiving power or receiving a reduced power). Preferably, only one of the voltage supply terminals (V_{DD} or ground) is power-gated, thereby reducing the speed and area penalties associated with the switch elements.

When the switch elements are controlled to de-couple the associated programmable logic blocks from the associated supply voltage, these programmable logic blocks are effectively disabled, thereby dramatically reducing the static power consumption of these blocks.

[0030] Fig. 3 is a block diagram of a conventional PLD 300 having four programmable logic blocks 301-304, which are all powered by the same off-chip V_{DD} voltage supply 305. Note that all four programmable logic blocks 301-304 are coupled to receive the V_{DD} supply voltage during normal operating conditions, even if some of these blocks are not used in the circuit design.

[0031] Fig. 4 is a block diagram of a PLD 400 in accordance with one embodiment of the present invention. Similar elements in Figs. 3 and 4 are labeled with similar reference numbers. Thus, PLD 400 includes programmable logic blocks 301-304 and V_{DD} voltage supply 305. In addition, PLD 400 includes switch elements 401-408, and control circuit 409. In the described embodiment, switch elements 401-404 are implemented by PMOS power-gating transistors 451-454, respectively, and switch elements 405-408 are implemented by NMOS power-gating transistors 455-458, respectively. In other embodiments, switch elements 401-408 may be any switch known to those ordinarily skilled in the art. Control circuit 409 is implemented by inverters 411-414, NOR gates 421-424, configuration memory cells 431-434, and user logic input terminals 441-444.

[0032] NOR gates 421-424 and inverters 411-414 are configured to generate power-gating control signals $SLEEP_1$ - $SLEEP_4$ and $SLEEP\#_1$ - $SLEEP\#_4$, in response to the configuration data values CD_1 - CD_4 stored in configuration memory cells 431-434, respectively, and the user control signals UC_1 - UC_4 provided on user logic input terminals 441-444, respectively.

[0033] For example, NOR gate 421 is coupled to receive configuration data value CD_1 from configuration memory cell 431 and user control signal UC_1 from user logic input terminal 441. If either the configuration data value CD_1 or the user control signal UC_1 is activated to a logic high state, then NOR gate 421

provides an output signal ($SLEEP\#_1$) having a logic "0" state. In response, inverter 411, which is coupled to the output terminal of NOR gate 421, provides an output signal ($SLEEP_1$) having a logic "1" state.

[0034] The $SLEEP_1$ signal is applied to the gate of PMOS power-gating transistor 451, which is coupled between block 301 and the V_{DD} voltage supply terminal. The $SLEEP\#_1$ signal is applied to the gate of NMOS power-gating transistor 455, which is coupled between block 301 and the ground voltage supply terminal. The logic "0" state of the $SLEEP\#_1$ signal causes NMOS power-gating transistor 455 to turn off, thereby de-coupling block 301 from the ground supply voltage terminal. Similarly, the logic "1" state of the $SLEEP_1$ signal causes PMOS power-gating transistor 451 to turn off, thereby de-coupling block 301 from the V_{DD} supply voltage terminal. De-coupling block 301 from the V_{DD} and ground supply voltage terminals effectively disables block 301, thereby minimizing the static leakage current in this block.

[0035] If both the configuration data value CD_1 , and the user control signal UC_1 , are de-activated to a logic low state, then NOR gate 421 provides a $SLEEP\#_1$ signal having a logic "1" state, and inverter 411 provides a $SLEEP_1$ signal having a logic "0" state. The logic "1" state of the $SLEEP\#_1$ signal causes NMOS power-gating transistor 455 to turn on, thereby coupling block 301 to the ground supply voltage terminal. Similarly, the logic "0" state of the $SLEEP_1$ signal causes PMOS power-gating transistor 451 to turn on, thereby coupling block 301 to the V_{DD} supply voltage terminal. Coupling block 301 to the V_{DD} and ground supply voltage terminals effectively enables block 301.

[0036] Programmable logic block 302 may be enabled and disabled in response to configuration data value CD_2 , and user control signal UC_2 , in the same manner as block 301. Similarly, programmable logic block 303 may be enabled and disabled in response to configuration data value CD_3 , and user control signal UC_3 , in the same manner as block 301. Programmable logic block 304 may be enabled and disabled in response to configuration

data value CD_4 and user control signal UC_4 , in the same manner as block 301.

[0037] As described above, when a programmable logic block is used and active, the associated power-gating transistors are turned on. Conversely, when a programmable logic block is unused or inactive, the associated power gating transistors are turned off. The $SLEEP_1$ - $SLEEP_4$ and $SLEEP\#_1$ - $SLEEP\#_4$ signals can be controlled by the configuration data values CD_1 - CD_4 stored by configuration memory cells 431-434, which are best suited for disabling the associated blocks at design time. If a block is not disabled at design time, this block can be disabled at run time by the user control signals UC_1 - UC_4 , which may be generated by the user logic, or by other means.

[0038] In accordance with another embodiment of the present invention, some blocks have multiple supply voltages. In this case all of the supply rails should be power-gated to achieve maximum power reduction. In accordance with another embodiment, only one switch element may be associated with each block. That is, the blocks are power-gated by de-coupling the block from only one power supply terminal, and not both the V_{DD} and ground supply voltage terminals, thereby conserving layout area.

[0039] The granularity of the power-gated programmable logic blocks can range from arbitrarily small circuits to significant portions of the PLD. The decision concerning the size of each programmable logic block is made by determining the desired trade-off between power savings, layout area overhead of the switch elements and the control circuit, and speed penalty. In a FPGA, each programmable logic block may be selected to include one or more configuration logic blocks (CLBs), input/output blocks (IOBs), and/or other resources of the FPGA (such as block RAM, processors, multipliers, adders, transceivers).

[0040] Another way to disable a programmable logic block is by scaling down the local supply voltage to the block as low as possible, which dramatically reduces the power consumption, both static and dynamic, of the block. To scale down the local supply voltage in this manner, each independently controlled

programmable logic block is powered by a separate switching regulator.

[0041] Fig. 5 is a block diagram of a PLD 500 that implements switching regulators in accordance with one embodiment of the present invention. Similar elements in Figs. 3 and 5 are labeled with similar reference numbers. Thus, PLD 500 includes programmable logic blocks 301-304 and V_{DD} voltage supply 305. In addition, PLD 500 includes switching regulators 501-504, which are coupled between blocks 301-304, respectively, and V_{DD} voltage supply 305. Switching regulators 501-504 are controlled by control circuits 511-514, respectively. In the described embodiment, switching regulators 501-504 reside on the same chip as blocks 301-304. However, in other embodiments, these switching regulators can be located external to the chip containing blocks 301-304. Switching regulators 501-504 can be programmably tuned to provide the desired supply voltages to the associated programmable logic blocks 301-304. For example, switching regulator 501 can provide a full V_{DD} supply voltage to programmable logic block 301 when this block is used and active. However, switching regulator 501 can further be controlled to provide a reduced voltage (e.g., some percentage of the V_{DD} supply voltage) to programmable logic block 301 when this block is unused or inactive. This reduced voltage may be predetermined (by design or via testing) depending on the desired circuit behavior. For example, this reduced voltage may be the minimum voltage required to maintain the state of the associated blocks. The power consumption of block 301 is significantly reduced when the supplied voltage is reduced in this manner.

[0042] Switching regulators 501-504 are controlled in response to the configuration data values C_1-C_4 stored in configuration memory cells 511-514, respectively, and the user control signals U_1-U_4 provided on user control terminals 521-524, respectively. A configuration data value (e.g., C_1) having an activated state will cause the associated switching regulator (e.g., switching regulator 501) to provide a reduced voltage to the associated programmable logic block (e.g., block 301).

Similarly, a user control signal (e.g., U_2) having an activated state will cause the associated switching regulator (e.g., switching regulator 502) to provide a reduced voltage to the associated programmable logic block (e.g., block 502). A configuration data value (e.g., C_3) and an associated user control signal (e.g., U_3) both having have deactivated states will cause the associated switching regulator (e.g., switching regulator 503) to provide the full V_{DD} supply voltage to the associated programmable logic block (e.g., block 503).

[0043] In accordance with one embodiment, configuration data values C_1-C_4 may be selected at design time, such that reduced voltages are subsequently applied to unused blocks during run time. User control signals U_1-U_4 may be selected during run time, such that reduced voltages are dynamically applied to inactive blocks at run time. Techniques for distributing multiple programmable down-converted voltages using on-chip switching voltage regulators are described in more detail in U.S. Patent Application Serial No. 10/606,619, "Integrated Circuit with High-Voltage, Low-Current Power Supply Distribution and Methods of Using the Same" by Bernard J. New et al., which is hereby incorporated by reference.

[0044] In the embodiment of Fig. 5, the granularity of the voltage scaled programmable logic blocks 301-304 should be fairly large because the overhead associated with switching regulators 501-504 is significant. In an FPGA, each programmable logic block 301-304 would most likely be divided into several clusters of configuration logic blocks (CLBs). The exact size of each programmable logic block is determined by the desired trade-off among power savings, layout area overhead of the switching regulators, and the speed penalty.

[0045] Fig. 6 is a block diagram of PLD 500, which shows switching regulators 501-504 in accordance with one embodiment of the present invention. Switching regulators 501-504 include control blocks 601-604, respectively, and high-voltage n-channel transistors 611-614, respectively. High-voltage n-channel transistors 611-614 can tolerate high voltages and may have relatively thick gate dielectric layers (e.g., 50 to 60

Angstroms) and relatively wide channel regions. In some embodiments, the gate dielectric thickness of the high-voltage n-channel transistors 611-614 is approximately 4 to 6 times thicker than the gate dielectric thickness used in the programmable logic blocks 301-304. The drain of each of n-channel transistors 611-614 is coupled to the V_{DD} voltage supply 305. The gates of n-channel transistors 611-614 are coupled to receive the control voltages $V_{c1}-V_{c4}$, respectively, from the corresponding control blocks 601-604. The sources of n-channel transistors 611-614 are configured to provide an operating voltage V_1-V_4 , respectively, to programmable logic blocks 301-304, respectively. The source of each n-channel transistor 611-614 is also coupled to the corresponding control block 601-604 in a feedback configuration.

[0046] Each of n-channel transistors 611-614 forms a power switch between the V_{DD} supply voltage 305 and the associated programmable logic block. Thick oxide n-channel transistors 611-614 are used to implement the power switches to ensure that a high voltage, herein referred to as V_{BOOST} , can be applied to the gates of n-channel transistors 611-614 when the associated programmable logic block is active. The high voltage V_{BOOST} increases the drive current of n-channel transistors 611-614. In accordance with one embodiment, the high voltage V_{BOOST} is about 2 to 2.5 times greater than V_{DD} . When the high voltage V_{BOOST} is applied to the gate of one of transistors 611-614, the corresponding operating voltage V_1-V_4 is pulled up to the full V_{DD} supply voltage.

[0047] When a programmable logic block (e.g., programmable logic block 301) is inactive, the associated operating voltage (e.g., V_1) is reduced. The operating voltage applied to the associated programmable logic block is preferably selected to be high enough to retain data stored in this programmable logic block. In one embodiment, the operating voltage is reduced to a voltage that is about one half the V_{DD} supply voltage. The operating voltage is reduced by applying a low voltage $V_{STANDBY}$ to the gate of the corresponding n-channel transistor (e.g.,

transistor 611). In one embodiment, the low voltage V_{STANDBY} is about 80 to 100 percent of the V_{DD} supply voltage.

[0048] In accordance with one embodiment, each of control blocks 601-604 is independently controlled to provide either the high voltage V_{BOOST} or the low voltage V_{STANDBY} to the associated n-channel transistor 611-614.

[0049] For example, control block 601 is configured to receive the user control signal U_1 and the configuration data value C_1 , which have been described above. If both the user control signal U_1 and the configuration data value C_1 are deactivated, then control block 601 provides a control voltage V_{c1} equal to the high voltage V_{BOOST} to the gate of n-channel transistor 611. As a result, an operating voltage V_1 equal to the V_{DD} supply voltage is applied to programmable logic block 301.

[0050] However, if either user control signal U_1 or configuration data value C_1 is activated, then control block 601 provides a control voltage V_{c1} equal to the low voltage V_{STANDBY} to the gate of n-channel transistor 611. As a result, an operating voltage V_1 approximately equal to one half the V_{DD} supply voltage is applied to programmable logic block 301.

[0051] To ensure that the operating voltage V_1 applied to programmable logic block 301 has a value of $\frac{1}{2} V_{\text{DD}}$ when the V_{STANDBY} voltage is applied to the gate of transistor 611, the control block 601 may include a feedback mechanism that adjusts the low voltage V_{STANDBY} signal until the operating voltage V_1 is precisely equal to $\frac{1}{2} V_{\text{DD}}$, or any other desired voltage.

[0052] It is well known that the gate current through a transistor typically increases by an order of magnitude for every 0.3 Volt increase in the V_{DD} supply voltage. It is therefore expected that reducing the operating voltage of a programmable logic block by half ($\frac{1}{2} V_{\text{DD}}$) will reduce the gate current through the transistors present in the programmable logic block by an order of magnitude or more. At the same time, the sub-threshold leakage of these transistors will also decrease with the reduced operating voltage. Based on earlier generation technology, the leakage current may be reduced by 70%

or more when reducing the operating voltage to $\frac{1}{2} V_{DD}$. Simulation of a ring oscillator shows that the ring oscillator will operate properly at the lower operating voltage ($\frac{1}{2} V_{DD}$). It can be expected the associated logic block will retain stored data using the lower operating voltage. Therefore, the proposed switching regulators are capable of achieving more than 70% reduction in leakage current without a significant increase in area penalty and without sacrificing desired functionality.

[0053] Although the invention has been described in connection with several embodiments, it is understood that this invention is not limited to the embodiments disclosed, but is capable of various modifications, which would be apparent to a person skilled in the art. For example, although the described embodiments included four programmable logic blocks, it is understood that other numbers of blocks can be used in other embodiments. Thus, the invention is limited only by the following claims.